

## **CLAIMS**

### **What is claimed is:**

1. A method comprising:

assigning an alias to each equivalence class of possibly overlapping memory accesses as defined by an alias analysis of an intermediate language program; and

defining a definition-use relationship between statements in each equivalence class wherein definition-statements which belong to the equivalence class reference the alias associated with that class and wherein use-statements which belong to the equivalence class reference the alias associated with that class.

2. The method of claim 1 further comprising constructing a dependence flow graph to represent said definition-use relationship comprising:

assigning a definition-node for each definition statement in the program;  
assigning a use-node for each use statement in the program;  
assigning an alias-node for each alias;  
introducing a single edge into the graph connecting each definition-node to its associated alias-node; and

introducing a single edge in the graph connecting each use-node to its associated alias-node.

3. The method of claim 1 further comprising first performing a memory alias analysis of said intermediate language program to partition the memory accesses of said intermediate language program into equivalence classes such that any two memory accesses that reference the same storage location belong to the same equivalence class.

4. The method of claim 2 further comprising performing a program analysis using said dependence flow graph.

5. The method of claim 4 wherein said program analysis comprises:

for each alias-node in the dependence flow graph assigning an initial value to the alias corresponding to said alias-node and adding said alias-node to a set of nodes; and

while said set of nodes is not empty, iteratively performing the following:

removing a node from said set of nodes;

if said node is an alias-node then adding the successors of said node in the dependence flow graph to said set of nodes;

if said node is a definition-node for a statement of the form PUT (A, E) then determining a value for E, updating said initial value based on the value of E; and adding A to said set of nodes.

6. The method of claim 5 wherein said initial value comprises a set of abstract values which form a join-complete partial order.

7. A machine-readable medium that provides instructions, which when executed by a processor, cause the processor to perform operations comprising:

assigning an alias to each equivalence class of possibly overlapping memory accesses as defined by an alias analysis of an intermediate language program; and

defining a definition-use relationship between statements in each equivalence class wherein definition statements which belong to the equivalence class reference the alias associated with that class and wherein used statement which belong to said equivalence class reference the alias associated with that class.

8. The machine-readable medium of claim 7, wherein said operations further comprise constructing a dependence flow graph to represent said definition-use relationship comprising:

assigning a definition-node for each definition statement in the program;

assigning a use-node for each use statement in the program;

assigning an alias-node for each alias;

introducing a single edge into the graph connecting each definition-node to its associated alias-node; and

introducing a single edge in the graph connecting each use-node to its associated alias-node.